**What is generative AI?**

Artificial Intelligence (AI) imitates human behavior by using machine learning to interact with the environment and execute tasks without explicit directions on what to output.

*Generative* AI describes a category of capabilities within AI that create original content. People typically interact with generative AI that has been built into chat applications. One popular example of such an application is [ChatGPT](https://openai.com/chatgpt), a chatbot created by OpenAI, an AI research company that partners closely with Microsoft.

Generative AI applications take in natural language input, and return appropriate responses in a variety of formats such as natural language, images, or code.

**Large language models**

Generative AI applications are powered by *large language models* (LLMs), which are a specialized type of machine learning model that you can use to perform *natural language processing* (NLP) tasks, including:

* Determining *sentiment* or otherwise classifying natural language text.
* Summarizing text.
* Comparing multiple text sources for semantic similarity.
* Generating new natural language.

While the mathematical principles behind these LLMs can be complex, a basic understanding of the architecture used to implement them can help you gain a conceptual understanding of how they work.

## Transformer models

Machine learning models for natural language processing have evolved over many years. Today's cutting-edge large language models are based on the transformer architecture, which builds on and extends some techniques that have been proven successful in modeling vocabularies to support NLP tasks - and in particular in generating language. Transformer models are trained with large volumes of text, enabling them to represent the semantic relationships between words and use those relationships to determine probable sequences of text that make sense. Transformer models with a large enough vocabulary are capable of generating language responses that are tough to distinguish from human responses.

Transformer model architecture consists of two components, or blocks:

* An encoder block that creates semantic representations of the training vocabulary.
* A decoder block that generates new language sequences.

In practice, the specific implementations of the architecture vary – for example, the Bidirectional Encoder Representations from Transformers (BERT) model developed by Google to support their search engine uses only the encoder block, while the Generative Pretrained Transformer (GPT) model developed by OpenAI uses only the decoder block.

While a complete explanation of every aspect of transformer models is beyond the scope of this module, an explanation of some of the key elements in a transformer can help you get a sense for how they support generative AI.

### Tokenization

The first step in training a transformer model is to decompose the training text into tokens - in other words, identify each unique text value. For the sake of simplicity, you can think of each distinct word in the training text as a token (though in reality, tokens can be generated for partial words, or combinations of words and punctuation).

For example, consider the following sentence:

*I heard a dog bark loudly at a cat*

To tokenize this text, you can identify each discrete word and assign token IDs to them. For example:

* I (1)
* heard (2)
* a (3)
* dog (4)
* bark (5)
* loudly (6)
* at (7)
* *("a" is already tokenized as 3)*
* cat (8)

The sentence can now be represented with the tokens: *[1 2 3 4 5 6 7 3 8]*. Similarly, the sentence "I heard a cat" could be represented as *[1 2 3 8]*.

As you continue to train the model, each new token in the training text is added to the vocabulary with appropriate token IDs:

* meow (9)
* skateboard (10)
* *and so on...*

With a sufficiently large set of training text, a vocabulary of many thousands of tokens could be compiled.

# What is Azure OpenAI?

Azure OpenAI Service is Microsoft's cloud solution for deploying, customizing, and hosting large language models. It brings together the best of OpenAI's cutting edge models and APIs with the security and scalability of the Azure cloud platform. Microsoft's partnership with OpenAI enables Azure OpenAI users to access the latest language model innovations.

Azure OpenAI supports many models that can serve different needs. These models include:

* **GPT-4 models** are the latest generation of generative pretrained (GPT) models that can generate natural language and code completions based on natural language prompts. Access to GPT-4 models is currently restricted - for access, existing Azure OpenAI customers can apply by [filling out this form](https://aka.ms/oai/get-gpt4).
* **GPT 3.5 models** can generate natural language and code completions based on natural language prompts. In particular, **GPT-35-turbo** models are optimized for chat-based interactions and work well in most generative AI scenarios.
* **Embeddings models** convert text into numeric vectors, and are useful in language analytics scenarios such as comparing text sources for similarities.
* **DALL-E models** are used to generate images based on natural language prompts. Currently, DALL-E models are in preview. DALL-E models aren't listed in the Azure OpenAI Studio interface and don't need to be explicitly deployed.

Models differ by speed, cost, and how well they complete specific tasks. You can learn more about the differences and latest models offered in the [Azure OpenAI Service documentation](https://learn.microsoft.com/en-us/azure/ai-services/openai/concepts/models).

In many cases, models can be used as-is. For example, in Azure OpenAI Service, you can deploy a GPT-4 model and immediately start using it from an application. However, you can also use an existing model as a foundational model - a starting point for further training with your own data. This approach is called fine-tuning, and it enables you to train a custom model that builds on the pre-trained model, but which is tuned to data that is relevant for your particular scenario. For example, a legal firm might fine-tune a model with the text from existing contracts and other proprietary legal documents to train a model that is optimized for generating contractual content.

## Azure AI Studio

Developers can work with these models in Azure AI Studio, a web-based environment where AI professionals can deploy, test, and manage LLMs that support generative AI app development on Azure.

# What are copilots?

Completed100 XP

* 4 minutes

The availability of LLMs has led to the emergence of a new software application, often referred to as a copilot. Copilots are often integrated into other applications and provide a way for users to get help with common tasks from a generative AI model. Copilots are based on a common architecture, so developers can build custom copilots for various business-specific applications and services.

You may see copilots appear within the products that you already use, for example, as a chat screen feature that opens up next to your file. These copilots use the content that is created or searched for in the product as specific information for its results.

It's helpful to think of how the creation of a large language model is related to the process of creating a copilot application:

1. A large amount of data is used to train a large language model.
2. Services such as Azure OpenAI Service make pretrained models available. Developers can use these pretrained models as they are, or fine-tune them with custom data.
3. Deploying a model makes it available for use in applications.
4. Developers can build copilots that submit prompts to models and generate content for use in applications.
5. Business users can use copilots to boost their productivity and creativity with AI-generated content.

Copilots have the potential to revolutionize the way we work. These copilots use generative AI to help with first drafts, information synthesis, strategic planning, and much more.

## Examples of copilots

Microsoft has already added copilots to commonly used applications.

In the Microsoft Edge browser, a copilot enables you to summarize the page you’re currently browsing or to generate new content.